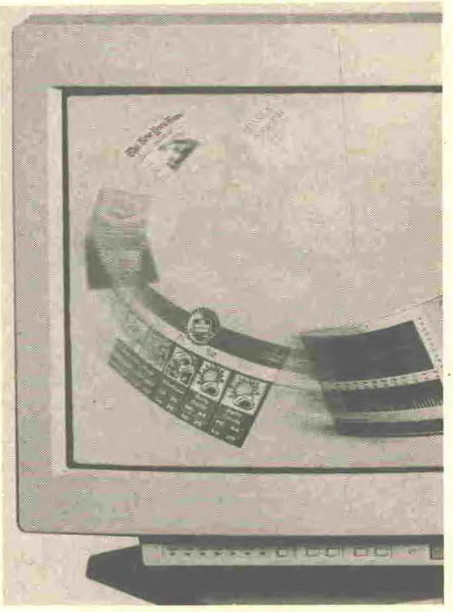
DATABASE

สิทธิศักดิ์ คล่องดี

การออกแบบฐานข้อมูล ตอนที่ 1



อธิบายถึงวิธีการออกแบบฐานข้อมูล ที่ถูกต้องตามหลักของความเป็นจริง และเน้นให้เห็นว่าการออกแบบที่ไม่ดีพอ จะมีปัญหาอะไรตามมา

สวัสดีครับ พบกันอีกครั้งหลังจากหายไปนาน เพราะผู้เขียนมีงานยุ่ง ก็เลยไม่ค่อยมีเวลาได้เขียนอะไรสนุก ๆ มาให้ผู้อ่านได้อ่านคลายเครียดหลังจากเลิกงาน สำหรับในครั้งนี้ผมจะมาพูดเรื่องราวเกี่ยวกับฐานข้อมูล (Database) ให้ท่านผู้อ่านที่สนใจในเรื่องฐานข้อมูลและคนที่ไม่สนใจก็สามารถอ่านได้นะครับ ไม่ผิดกติกา เราจะสังเกตเห็นว่าในโลกปัจจุบันนี้การใช้คอมพิวเตอร์ส่วนใหญ่ประมาณ 90 เปอร์เซ็นต์ จะต้องยุ่งเกี่ยวกับการจัดเก็บข้อมูล ถ้าหากว่าเป็นการจัดเก็บข้อมูลเล็ก ๆ น้อย ๆ เราก็จะนิยมเก็บไว้เป็นแฟ้มข้อมูลธรรมดา แต่ถ้าหากว่าข้อมูลมีจำนวนมากและมีความสัมพันธ์กันอย่างสลับซับซ้อนเราก็จะจัดเก็บในลักษณะของฐานข้อมูล และแน่นอนที่สุดว่า ในการออกแบบฐานข้อมูลเพื่อที่จะจัดเก็บข้อมูลที่มีเป็นจำนวนมากนั้น ถ้าหากมีการออกแบบอย่างดีก็จะทำให้การจัดเก็บ การค้นหา เป็นไปด้วยความรวดเร็ว ถูกต้อง แม่นยำ และยังช่วยลดความเสี่ยงที่จะเกิดขึ้นด้วย แต่ถ้าออกแบบไม่ดีปัญหาต่าง ๆ ก็จะตามมามากมายจนในที่สุดเราก็ไม่สามารถตามเช็ดถูปัดกวาดปัญหาเหล่านั้นได้ แล้วระบบที่ได้ทำการออกแบบลงไปนั้นก็จะถูกยกเลิกโดยปริยาย ทำให้เสียเวลาและค่าใช้จ่ายโดยได้รับผลตอบแทนกลับมาที่ไม่คุ้มค่า ซึ่งผู้เขียนได้เล็งเห็นความสำคัญของข้อมูลที่จะนำมาจัดเก็บ รวมถึงการออกแบบวิธีการจัดเก็บ ผู้เขียนจึงได้หยิบเรื่องของการออกแบบฐานข้อมูลมาเล่าให้ท่านผู้อ่านได้อ่านยามว่าง โดยผู้เขียนจะแบ่งเนื้อหาออกเป็นตอน ๆ ดังต่อไปนี้

1. การออกแบบฐานข้อมูล ซึ่งในตอนนี้จะได้อธิบายถึง วิธีการออกแบบฐานข้อมูลที่ถูกต้องตามหลักของความเป็นจริง และเน้นให้เห็นว่าการออกแบบฐานข้อมูลที่ไม่ดีพอจะมีปัญหาอะไรตามมา โดยจะแสดงปัญหาและวิธีการแก้ไขไว้อย่างชัดเจน

2. การสร้างฐานข้อมูลโดยใช้ Microsoft Access เป็นการแนะนำการสร้างฐานข้อมูลที่ได้ออกแบบในตอนที่ 1 โดยใช้ Microsoft Access เป็นตัวจัดการ ซึ่งในตอนนี้จะเน้นในเรื่อง การสร้างฐานข้อมูล การสร้าง Query การสร้าง Form และ Report

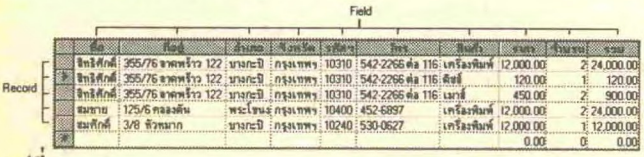
3. การออกแบบเพื่อใช้งานใน Microsoft Access เน้นในเรื่องของการประยุกต์ใช้งานจริงโดยใช้ Microsoft Access รวมถึงการเขียน Macro การแนะนำ Access Basic และระบบการรักษาความปลอดภัยใน Microsoft Access

4. การเขียนโปรแกรมเรียกใช้ฐานข้อมูลโดยใช้ Visual Basic จะเป็นการแนะนำการเขียนโปรแกรมติดต่อกับ Jet Engine โดยใช้ Visual Basic

สำหรับตอนนี้เป็นตอนที่ 1 เราจะมาเริ่มกันที่การศึกษาศัพท์ต่าง ๆ ที่เราจะใช้ในบทความนี้คือ

- **Primary Key (คีย์หลัก)** หมายถึง สิ่งที่จะใช้บ่งบอกถึงความแตกต่างในสิ่งที่เรากำลังจะพูดถึง เช่น ถ้าเราพูดถึงพนักงานในหน่วยงานต่าง Primary Key ก็จะหมายถึงรหัสพนักงานแต่ละคนซึ่งจะไม่ซ้ำกันเลย เมื่อเอ่ยถึงหมายเลขซึ่งเป็นรหัสของพนักงานหมายเลขใด ก็จะสามารถรู้ได้ทันทีว่าเป็นพนักงานคนใด

● **Table (ตาราง)** หมายถึง กลุ่มของข้อมูลที่จัดให้มีความสัมพันธ์กันในรูปของแนวนอนและแนวตั้ง ซึ่งในฐานข้อมูลแล้วเราจะเรียกข้อมูลที่เรียงกันเป็นแถวในแนวนอนว่า Record และจะเรียกข้อมูลที่วางในแนวตั้งว่า Field ดังรูปที่ 1



ชื่อ	ที่อยู่	อำเภอ	จังหวัด	รหัสฯ	โทร	สินค้า	ราคา	จำนวน	รวม
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	เครื่องพิมพ์	12,000.00	2	24,000.00
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	ดิสก์	120.00	1	120.00
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	เมาส์	450.00	2	900.00
สมชาย	125/6 คลองตัน	พระโขนง	กรุงเทพฯ	10400	452-6897	เครื่องพิมพ์	12,000.00	2	24,000.00
สมศักดิ์	3/8 หัวหมาก	บางกะปิ	กรุงเทพฯ	10240	530-0627	เครื่องพิมพ์	12,000.00	1	12,000.00
							0.00	0	0.00

รูปที่ 1

● **Relation (ความสัมพันธ์)** หมายถึง สิ่งที่เกี่ยวข้องกันของสิ่งของสองสิ่ง เช่น คนกับบ้าน ซึ่งสองสิ่งนี้จะมีความสัมพันธ์กัน แต่ถ้าเป็นความสัมพันธ์ทาง Relational Database แล้วจะหมายถึง ตาราง 2 ตารางที่มีความสัมพันธ์กัน และความสัมพันธ์ยังแบ่งออกได้เป็น 4 อย่างคือ

1. One-To-One เช่น ความสัมพันธ์ระหว่างทะเบียนรถยนต์กับรถยนต์ จะเป็นแบบ One-To-One เพราะรถ 1 คันจะมีทะเบียนหมายเลขเดียว และทะเบียน 1 หมายเลขจะมีรถเพียงคันเดียวเท่านั้นที่อ้างถึง

2. One-To-Many เช่น บริษัท 1 บริษัทมีพนักงานอยู่หลายคน

3. Many-To-One เช่น พนักงานหลายคนทำงานใน 1 บริษัท

4. Many-To-Many เช่น นักศึกษา 1 คนลงทะเบียนหลายวิชา และวิชาแต่ละวิชาก็ถูกนักศึกษาหลายคนลงทะเบียนเช่นเดียวกัน

เมื่อผู้อ่านได้ทำความเข้าใจเกี่ยวกับศัพท์ต่าง ๆ แล้ว เราจะมาทำความเข้าใจให้ตรงกันเกี่ยวกับขอบเขตของฐานข้อมูลที่เราจะออกแบบ สมมติว่าท่านได้รับมอบหมายงานให้ออกแบบโปรแกรมเพื่อมาใช้เก็บข้อมูลเกี่ยวกับการจัดเก็บใบสั่งซื้อให้กับองค์กร โดยโปรแกรมที่ออกแบบจะต้องมีความสามารถดังต่อไปนี้

1. สามารถจัดเก็บรายละเอียดของลูกค้าได้
2. สามารถเก็บรายการสินค้าที่จำหน่ายได้
3. สามารถเก็บรายการสั่งซื้อของลูกค้าแต่ละคนได้
4. สามารถค้นหาใบสั่งซื้อแต่ละใบได้
5. สามารถค้นหาลูกค้าแต่ละคนว่าได้ซื้ออะไรไปบ้าง
6. สามารถพิมพ์ใบสั่งสินค้าออกทางเครื่องพิมพ์ได้

ถ้าสมมติว่าความต้องการของระบบมีเพียงเท่านี้ เราก็จะเริ่มทำการออกแบบฐานข้อมูล โดยการออกแบบฐานข้อมูลในที่นี้จะใช้ Relational Database ซึ่งจะแบ่งเป็นขั้นตอนคร่าว ๆ ดังต่อไปนี้

1. จากเงื่อนไขต่าง ๆ ที่กล่าวมาข้างต้นเรามาดูว่าจะต้องจัดเก็บอะไรข้างในฐานข้อมูลเพื่อที่จะใช้ในการตอบสนองต่อความต้องการ

2. นำข้อมูลที่ได้ออกแบบในข้อ 1 มาใส่ลงในตาราง (Relation Database จะมองข้อมูลที่เก็บเหมือนตาราง) และกำหนดคีย์ที่จะใช้ในการอ้างถึงข้อมูลในแต่ละ Record

3. ตรวจสอบความซ้ำซ้อนของข้อมูล และแก้ไขรายการที่ซ้ำซ้อนกัน โดยการ Minimize

4. เขียนโปรแกรมเพื่อเรียกใช้ข้อมูลในฐานข้อมูลที่ออกแบบ

เราจะเก็บอะไรบ้างในฐานข้อมูล สิ่งที่เราจะต้องเก็บก็คือสิ่งที่จะมาตอบสนองต่อความต้องการของผู้ใช้ โดยดูได้จากความต้องการ 6 ข้อที่กล่าวมา ซึ่งถ้าหากเราดูจากความต้องการแล้วทำให้เราพอสรุปคร่าว ๆ ได้ว่าควรจะเก็บสิ่งใดไว้ในฐานข้อมูล เช่น ชื่อลูกค้า ที่อยู่ อำเภอ จังหวัด รหัสไปรษณีย์ โทรศัพท์ ชื่อสินค้าที่สั่ง จำนวนที่สั่ง ราคาต่อหน่วย จำนวนเงิน เป็นต้น ทำไมต้องเก็บสิ่งเหล่านี้ไว้ในฐานข้อมูล รู้ได้อย่างไรว่าจะต้องเก็บสิ่งเหล่านี้ คำตอบก็คือ สิ่งเหล่านี้เป็นสิ่งที่ผมจะยกตัวอย่างการพิจารณาให้ท่านผู้อ่านทราบ เช่น ความต้องการข้อที่ 1 บอกไว้ว่า "สามารถเก็บรายละเอียดของลูกค้าได้" ซึ่งลูกค้าก็หมายถึงบุคคลหรือองค์กร และสิ่งที่เราจะต้องเก็บไว้ในฐานข้อมูลก็คือ ชื่อบุคคล (ลูกค้า) หรือชื่อองค์กรนั่นเอง นอกจากชื่อแล้วเราก็จะเก็บรายละเอียดที่เป็นส่วนประกอบของชื่อบุคคลหรือชื่อองค์กรก็ควรจะเก็บด้วย เช่น ที่อยู่ อำเภอ จังหวัด ฯลฯ สิ่งเหล่านี้เก็บไว้เพื่อที่ว่าถ้าหากเราต้องการจะติดต่อจะได้ติดต่อได้ สำหรับรายละเอียดอื่น ๆ ที่จะเก็บก็ให้ผู้อ่านพิจารณาเป็นข้อ ๆ ตามความต้องการดังตัวอย่างที่กล่าวมาข้างต้น ดังนั้นเมื่อเราได้พิจารณาครบทุกข้อของความต้องการแล้วเราก็นำความต้องการเหล่านี้มาสร้างเป็นตารางตามลักษณะของ Relational Database แล้วจะได้ตารางดังรูปที่ 2 (จงจำไว้อย่างว่า "*ยิ่งความต้องการมาก รายละเอียดที่ต้องเก็บลงฐานข้อมูลก็ยิ่งมากตามมาด้วย*")

จากรูปที่ 2 เราสามารถเก็บข้อมูลได้ครบทุกอย่าง แต่เป็นการออกแบบที่ไม่ดีเพราะมีข้อจำกัดดังต่อไปนี้คือ (ปัญหาที่ 1)

ชื่อ	ที่อยู่	อำเภอ	จังหวัด	รหัสฯ	โทร	สินค้า	ราคา	จำนวน	รวม
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	เครื่องพิมพ์	12,000.00	2	24,000.00
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	ดิสก์	120.00	1	120.00
สิทธิศักดิ์	355/76 ซอยพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116	เมาส์	450.00	2	900.00
สมชาย	125/6 คลองตัน	พระโขนง	กรุงเทพฯ	10400	452-6897	เครื่องพิมพ์	12,000.00	2	24,000.00
สมศักดิ์	3/8 หัวหมาก	บางกะปิ	กรุงเทพฯ	10240	530-0627	เครื่องพิมพ์	12,000.00	1	12,000.00
							0.00	0	0.00

รูปที่ 2

1. เมื่อสิทธิศักดิ์สั่งซื้อสินค้า 3 ชนิด (เครื่องพิมพ์ ดิสก์ เม้าส์) จะสังเกตเห็นว่าต้องพิมพ์ชื่อและที่อยู่ของสิทธิศักดิ์ซ้ำกันเท่ากับจำนวนสินค้าที่สั่ง ซึ่งจะทำให้เสียเวลาและโอกาสที่ผู้ป้อนข้อมูลจะพิมพ์ผิดมีมาก

2. เมื่อสิทธิศักดิ์เปลี่ยนชื่อหรือที่อยู่ จะต้องตามแก้ไขเท่ากับจำนวนสินค้าที่สิทธิศักดิ์สั่งซื้อ ถ้าแก้ไขไม่ครบจะทำให้ข้อมูลขัดแย้งกัน เช่น สิทธิศักดิ์เปลี่ยนชื่อเป็นสิทธิพงศ์ ถ้าแก้ไขแต่รายการที่ 1 และรายการที่ 2 รายการที่ 3 ไม่ได้แก้ไข จะสังเกตเห็นว่าสิทธิพงศ์สั่งซื้อสินค้า 2 อย่าง และมีสิทธิศักดิ์สั่งซื้อ 1 อย่าง ซึ่งจริง ๆ แล้วสิทธิศักดิ์และสิทธิพงศ์เป็นคน ๆ เดียวกัน ตรงจุดนี้จะทำให้เกิดการขัดแย้งกันของข้อมูลเกิดขึ้น ซึ่งจะทำให้ข้อมูลในฐานข้อมูลของเราไม่ถูกต้องตามหลักความเป็นจริง

3. สิทธิศักดิ์ สมชาย สมศักดิ์ สั่งซื้อสินค้าชนิดเดียวกันคือเครื่องพิมพ์ ผู้ที่ทำหน้าที่ป้อนข้อมูลต้องพิมพ์คำว่า "เครื่องพิมพ์" ซ้ำกันเท่ากับจำนวนรายการที่สั่ง

4. ไม่สามารถแยกได้ว่าสิทธิศักดิ์สั่งซื้อสินค้า 3 ชนิดในครั้งเดียวกันหรือสั่งหลายครั้ง

5. เมื่อมีลูกค้ามาติดต่อแล้วลูกค้าต้องซื้อสินค้าทุกครั้งจึงสามารถพิมพ์ลงฐานข้อมูลได้

6. เมื่อมีรายการสินค้าที่จะขายใหม่จะพิมพ์ลงฐานข้อมูลได้เมื่อมีลูกค้าสั่งซื้อสินค้า

จากปัญหาที่กล่าวมานี้ปัญหาหลัก ๆ ก็คือข้อมูลเกิดการซ้ำซ้อนกันมากเกินไป เพราะเราต้องพิมพ์ข้อมูลซ้ำ ๆ กันหลายครั้ง ทำให้เสียเวลาและโอกาสที่ผู้พิมพ์จะพิมพ์ผิดก็มีมาก การแก้ปัญหาดังกล่าวคือเราต้องแยกตารางข้างบนออกเป็นตารางย่อย ๆ เราเรียกวิธีการนี้ว่าการ Minimize เพื่อลดความซ้ำซ้อนของข้อมูล ทำให้ง่ายต่อการใช้งาน และช่วยลดความผิดพลาดที่จะเกิดขึ้น โดยการ Minimize มีหลักการคร่าว ๆ ก็คือ ให้นำเอาข้อมูลที่ขึ้นตรงต่อกันไปไว้รวมกัน เช่น ชื่อ ที่อยู่ ของลูกค้าควรจะเป็น 1 ตาราง รายชื่อสินค้าและราคาควรจะอยู่อีก 1 ตาราง เป็นต้น ซึ่งเมื่อผ่านขั้นตอนการ Minimize แล้วเราจะได้ตารางดัง 3 ตารางในรูปที่ 3

ลูกค้า

	ชื่อ	ที่อยู่	อำเภอ	จังหวัด	รหัสฯ	โทร
▶	สิทธิศักดิ์	355/76 ลาดพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116
	สมชาย	125/6 คลองตัน	พระโขนง	กรุงเทพฯ	10400	452-6897
	สมศักดิ์	3/8 หัวหมาก	บางกะปิ	กรุงเทพฯ	10240	530-0627
	สมหญิง	45/98 วังนา	เมือง	เชียงใหม่	45000	458-987
*						

รูปที่ 3

สินค้า

	สินค้า	ราคา
▶	เครื่องพิมพ์	12,000.00
	ดิสซ์	120.00
	เมาส์	450.00
	CD-ROM	950.00
*		0.00

รายการสั่งซื้อ

	ชื่อ	สินค้า	จำนวน	ราคา
▶	สิทธิศักดิ์	เครื่องพิมพ์	2	24,000.00
	สิทธิศักดิ์	ดิสซ์	1	120.00
	สิทธิศักดิ์	เมาส์	2	900.00
	สมชาย	เครื่องพิมพ์	2	24,000.00
	สมศักดิ์	เครื่องพิมพ์	1	12,000.00
*			0	0.00

เมื่อทำการ Minimize แล้วจะสังเกตเห็นว่าเราได้ตารางใหม่เกิดขึ้น โดยตารางใหม่ที่เกิดขึ้นนี้จะเป็นการนำ Field ที่อยู่ในตารางแรกมาแบ่งออกเป็นตารางย่อย ๆ จะได้ทั้งหมด 3 ตาราง คือ

1. ตารางลูกค้า เอาไว้เก็บรายละเอียดของลูกค้า

2. ตารางสินค้า เอาไว้เก็บรายการสินค้าที่มีจำหน่าย

3. ตารางรายการสั่งซื้อ เอาไว้เก็บรายการสั่งซื้อต่าง ๆ

จากตารางทั้ง 3 ตารางที่สร้างขึ้นมาใหม่จะช่วยแก้ปัญหาที่ 1 ที่ได้กล่าวมาแล้วในรูปที่ 2 ดังต่อไปนี้คือ

1. สามารถเพิ่มลูกค้าเข้าในฐานข้อมูลไว้ก่อนโดยที่ลูกค้ายังไม่จำเป็นต้องสั่งสินค้า โดยพิมพ์รายการของลูกค้าลงในตารางลูกค้า เช่น ลูกค้าที่ชื่อสมหญิง จะมีรายชื่อในฐานข้อมูลแต่ยังไม่ได้สั่งซื้อสินค้า สังเกตได้จากในตารางรายการสั่งซื้อยังไม่มีชื่อของสมหญิง

2. สามารถเพิ่มรายการสินค้าเข้ามาขายใหม่ โดยไม่จำเป็นต้องมีลูกค้าสั่ง สังเกตจากตารางสินค้า มีสินค้าตัวใหม่คือ CD-ROM แต่ยังไม่มีรายการสั่งซื้อ สังเกตได้จากตารางรายการสั่งซื้อ

3. การสั่งซื้อสินค้าแต่ละครั้งไม่ต้องพิมพ์ข้อมูลซ้ำกันมาก ๆ เหมือนในตารางในรูปที่ 2 เพราะรายการสั่งซื้อจะพิมพ์ลงในตารางรายการสั่งซื้อ ซึ่งจะพิมพ์แค่ ชื่อผู้สั่ง สินค้าที่สั่ง จำนวนที่สั่ง และราคารวม จะช่วยให้เสียเวลาในการพิมพ์ข้อมูลลงในฐานมูลน้อยลงกว่าเดิม

แต่ตารางทั้ง 3 ตารางนี้ยังใช้ได้ไม่ดีเท่าที่ควรเพราะยังมีปัญหาต่าง ๆ ดังต่อไปนี้ (ปัญหาที่ 2)

1. ไม่สามารถแยกได้ว่า รายการที่สั่งซื้อที่มีในตารางรายการสั่งซื้อนั้นมีใบสั่งซื้อกี่ใบ เช่น รายการสั่งซื้อของสิทธิศักดิ์ ที่สั่งสินค้า 3 ชนิด เป็นการสั่งซื้อในครั้งเดียวหรือเป็นการสั่งหลายครั้ง

2. เมื่อมีลูกค้าคนใหม่มาติดต่อกับบริษัท และลูกค้าคนใหม่มีชื่อ-นามสกุล และที่อยู่ เหมือนกันกับสมศักดิ์ (โอกาสเป็นไปได้ที่คนเราจะมีชื่อและนามสกุลเหมือนกัน และที่อยู่ก็มีโอกาสจะซ้ำกันได้ถ้าหากลูกค้า 2 คนอาศัยอยู่ในอาพาร์ทเมนท์เดียวกัน) ถ้าหากเราป้อนข้อมูลลงฐานข้อมูลก็จะทำให้เรามี Record 2 Record ที่ซ้ำกัน ซึ่งเราไม่สามารถรู้ได้เลยว่าเป็นสมศักดิ์คนไหน เมื่อป้อนลงฐานข้อมูลแล้ว

3. เมื่อสิทธิศักดิ์เปลี่ยนชื่อ ต้องตามเข้าไปแก้ในตาราง 2 ตาราง คือตารางลูกค้าและตารางรายการสั่งสินค้า ซึ่งถ้าหากแก้ไม่หมดก็จะทำให้เกิดความขัดแย้งของข้อมูลขึ้น

4. เมื่อเปลี่ยนรายชื่อสินค้าจากดิสค์เป็นดิสเกตต์ ต้องเข้าไปแก้ถึง 2 ตาราง คือตารางสินค้าและตารางรายการสั่งสินค้า

การแก้ปัญหาทั้ง 3 ตารางทำได้โดยการ Minimize ตารางออกมาใหม่ โดยเพิ่มคุณสมบัติบางอย่างเข้าไป ซึ่งจะได้ดังตารางในรูปที่ 4

ลูกค้า

	รหัสลูกค้า	ชื่อ	ที่อยู่	อำเภอ	จังหวัด	รหัสฯ	โทร
▶	1	สิทธิศักดิ์	355/76 ลาดพร้าว 122	บางกะปิ	กรุงเทพฯ	10310	542-2266 ต่อ 116
	2	สมชาย	125/6 คลองตัน	พระโขนง	กรุงเทพฯ	10400	452-6897
	3	สมศักดิ์	3/8 หัวหมาก	บางกะปิ	กรุงเทพฯ	10240	530-0627
	4	สมหญิง	45/98 วังนา	เมือง	เชียงใหม่	45000	458-987
*	(AutoNumber)						

รูปที่ 4

สินค้า

	รหัสสินค้า	สินค้า	ราคา
▶	1	เครื่องพิมพ์	12,000.00
	2	ดิสซ์	120.00
	3	เมาส์	450.00
*	(AutoNumber)		0.00

รายการสั่งซื้อ

	รหัสรายการสั่งซื้อ	รหัสลูกค้า	รหัสสินค้า	จำนวน	วันที่สั่งซื้อ
▶	1	1	1	2	1/4/97
	1	1	2	1	1/4/97
	2	1	3	2	5/4/97
	3	2	1	2	1/4/97
	4	2	1	1	1/4/97
	5	3	3	2	8/5/97
*	0	0	0	0	

มีสิ่งที่เปลี่ยนแปลงเกิดขึ้นกับตารางทั้ง 3 ตาราง โดยทุก ๆ ตารางจะมี Field รหัสเกิดขึ้นมาเพื่อแยกความแตกต่างในแต่ละ Record (ยกเว้นตารางรายการสั่งซื้อ รหัสใบสั่งซื้อจะไม่ใช่ Field ที่จะมาแยกความแตกต่างในแต่ละ Record เอาไว้ผู้เขียนจะอธิบายในตอนท้าย) เราจะเรียกสิ่งที่มาแยกความแตกต่างนี้ว่า Primary Key (คีย์หลัก) และในตารางรายการสั่งซื้อนั้นจะเก็บรหัสลูกค้าและชื่อสินค้าแทนชื่อจริงเหมือนในตารางในรูปที่ 3 และในตารางรายการสั่งซื้อนี้ยังได้เพิ่ม Field "วันที่สั่งซื้อ" ด้วย แต่เราจะตัด Field "รวม" ออก เพราะ Field นี้ไม่จำเป็นต้องมี เราสามารถเขียนโปรแกรมคำนวณเองได้ในขณะทำงาน

การทำงานของตารางในรูปที่ 4 จะเป็นดังนี้คือ การเพิ่มรายชื่อลูกค้าและสินค้าใหม่ทุก ๆ ครั้งจะต้องมีการให้รหัสลูกค้าเสมอ และการสั่งซื้อสินค้าทุกครั้งต้องมีรหัสใบสั่งซื้อ รหัสลูกค้า รหัสสินค้า ดังตัวอย่าง เช่น (ดูรูปที่ 4 ประกอบ)

ใบสั่งซื้อหมายเลข 1 จะมี 2 Record แต่มีรหัสลูกค้าที่สั่งซื้อคือ 1 (สิทธิศักดิ์) และรหัสสินค้าที่สั่งมีอยู่ 2 ชนิดคือ 1 (เครื่องพิมพ์) และ 2 (ดิสค์) โดยสั่งซื้อสินค้าชนิดที่ 1 จำนวน 2 ชิ้น และสินค้าชนิดที่ 2 จำนวน 1 ชิ้น ซึ่งอธิบายได้ว่า เมื่อวันที่ 1/4/97 สิทธิศักดิ์ได้ซื้อสินค้า 2 รายการคือ เครื่องพิมพ์ 2 เครื่อง และดิสค์ 1 กล่อง โดยมีรหัสใบสั่งซื้อเท่ากับ 1

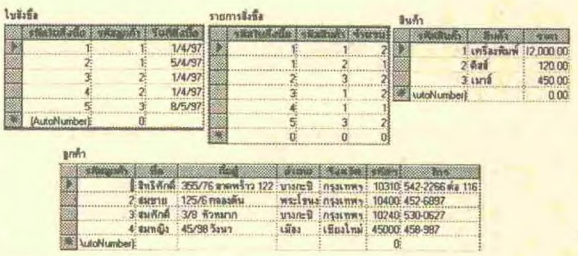
ดูจาก Field รหัสลูกค้าในตารางรายการสั่งซื้อ จะพบว่ามี 3 Record ที่มีรหัสลูกค้าเหมือนกัน ซึ่ง 2 Record แรกได้อธิบายไปแล้ว ส่วน Record ที่ 3 จะมีความหมายดังนี้คือ สิทธิศักดิ์ซื้อสินค้าชื่อเม้าส์ จำนวน 2 ตัว เมื่อวันที่ 5/4/97 โดยมีรหัสใบสั่งซื้อเท่ากับ 2 (สำหรับ Record อื่น ๆ ลองดูกันเองนะครับว่ามีความหมายอย่างไร)

จากตารางในรูปที่ 4 3 ตารางนี้ทำให้แก้ปัญหาที่ 2 (ถ้าจำไม่ได้ให้กลับไปอ่านปัญหาที่ 2 ก่อนที่เราจะทำการ Minimize ตารางในรูปที่ 4)

1. จากตารางรายการสั่งซื้อ บอกให้เราทราบว่ามีใบสั่งซื้อ 5 ใบ (ดูจากหมายเลขใบสั่งซื้อ 1, 2, 3, 4, 5) นอกจากนี้ยังสามารถบอกได้ว่าลูกค้าแต่ละคนมีการสั่งซื้อสินค้ากี่ครั้ง โดยดูเปรียบเทียบระหว่าง Field รหัสใบสั่งซื้อ กับ Field รหัสลูกค้า ถ้าหากซ้ำกัน แสดงว่าเป็นการสั่งซื้อครั้งเดียวกัน เช่น รหัสใบสั่งซื้อเท่ากับ 1 และรหัสลูกค้าเท่ากับ 1 มีอยู่ 2 Record ซึ่งรายการสั่งซื้อที่ซ้ำกันนี้จะเป็นการสั่งซื้อในครั้งเดียวกัน (ใบสั่งซื้อใบเดียวกัน) ถ้ามองดูใน Record ต่อมา จะเห็นรหัสใบสั่งซื้อเท่ากับ 2 แต่ยังคงมีรหัสลูกค้าที่สั่งซื้อเท่ากับ 1 เหมือนเดิม ดังนั้นเราสรุปได้ว่า ลูกค้าที่มีรหัสเท่ากับ 1 (สิทธิศักดิ์) มีการใช้รหัสใบสั่งซื้อ 2 รหัส ซึ่งจะหมายถึงใช้ใบสั่งซื้อ 2 ใบ แสดงว่ามีการสั่งซื้อ 2 ครั้ง โดยใบสั่งซื้อใบไหนสั่งก่อน-หลังดูได้จาก Field วันที่สั่งซื้อ และถ้าอยากจะรู้ว่าในใบสั่งซื้อแต่ละใบซื้อสินค้าชนิดใดบ้าง ดูได้จาก Field รหัสสินค้า

2. เมื่อมีการเปลี่ยนชื่อของสิทธิศักดิ์เป็นสิทธิพงศ์ จะไปเปลี่ยนเพียงตารางเดียวเท่านั้นคือตารางลูกค้า และเช่นเดียวกัน ถ้าต้องการเปลี่ยนชื่อสินค้าจากดิสค์เป็นดิสเกตต์ ก็จะไปเปลี่ยนเพียงตารางเดียวเท่านั้นคือตารางสินค้า ซึ่งจะช่วยลดเวลาในการแก้ไขและลดความผิดพลาดที่จะเกิดเพราะเราไม่ต้องเข้าไปแก้ไขในหลายตารางเหมือนในปัญหาที่ 2

จากรูปที่ 4 นี้ยังไม่ใช่ตารางที่ดีที่สุดเพราะยังมีการซ้ำซ้อนกันของข้อมูลอยู่ ให้สังเกตดูฟิลด์ รหัสใบสั่งซื้อ รหัสลูกค้า วันที่สั่งซื้อ ทั้ง 3 ฟิลด์นี้จะมีข้อมูลที่ซ้ำซ้อนกันเท่ากับจำนวนสินค้าที่สั่งซื้อ ถ้าหากคน 1 คนสั่งซื้อสินค้าหลาย ๆ อย่าง จะเห็นว่ามีรหัสลูกค้า รหัสใบสั่งซื้อ วันที่สั่งซื้อสินค้าซ้ำกันกับจำนวนสินค้าที่สั่งซื้อ ดังนั้นเราจึงทำการ Minimize ตารางออกไปเป็นตารางต่าง ๆ ได้ 4 ตาราง ดังรูปที่ 5



รูปที่ 5

จากลักษณะการใช้งานในตารางรายการสั่งซื้อจะสังเกตเห็นว่าตารางรายการสั่งซื้อจะใช้รหัสสินค้าในตารางรายการสั่งสินค้า เพื่อไปค้นหาชื่อและราคาสินค้าในตารางสินค้า และจะใช้รหัสลูกค้าในตารางใบสั่งซื้อไปค้นหาลูกค้าในตารางลูกค้า ซึ่งการแสดงคุณสมบัติดังนี้จะหมายถึงความสัมพันธ์ (Reletion) ที่มีขึ้นในแต่ละตาราง โดยที่ตารางรายการสั่งซื้อจะเป็นตารางกลางคอยเชื่อมต่อระหว่างตารางลูกค้าและตารางสินค้า และนอกจากนี้เรายังสังเกตได้ว่า ถ้าหากไม่มีลูกค้ารายใดเลยสั่งสินค้า ที่ตารางรายการสั่งสินค้าก็จะไม่มีข้อมูลเกิดขึ้น และการที่ตารางแต่ละตารางจะสามารถเชื่อมโยงหรืออ้างถึงกันได้นั้นต้องอาศัย Primary Key โดยตารางแต่ละตารางจะมี Field ที่ทำเป็น Primary Key ดังตารางที่ 1

เมื่อตารางแต่ละตารางมีความสัมพันธ์กันแล้ว เราจะมาพิจารณาว่า แต่ละตารางมีความสัมพันธ์กันแบบใด ซึ่งมีทั้งหมด 4 แบบตามที่กล่าวมาแล้วก่อนหน้านี้ รูปที่ 6 แสดงภาพของความสัมพันธ์ประกอบ

จากรูปที่ 6 สามารถอธิบายได้ดังตารางที่ 2

รูปที่ 6

ตารางที่ 1

ชื่อตาราง	Primary Key	หมายเหตุ
ลูกค้า	รหัสลูกค้า	
สินค้า	รหัสสินค้า	
ใบสั่งซื้อ	รหัสใบสั่งซื้อ	
รายการสั่งซื้อ	รหัสใบสั่งซื้อ รหัสสินค้า	ทั้ง 2 Field ทำหน้าที่รวมกันเป็น Primary Key

ตารางที่ 2

ความสัมพันธ์	จาก		ไป		คำอธิบายและเหตุผล
	ตาราง	ฟิลด์	ตาราง	ฟิลด์	
One-To-Many	ลูกค้า	รหัสลูกค้า	ใบสั่งซื้อ	รหัสลูกค้า	เพราะว่าลูกค้า 1 คนสามารถสั่งซื้อสินค้าได้หลายครั้ง และการสั่งซื้อแต่ละครั้งก็สามารถซื้อ เราจึงเรียกว่าความสัมพันธ์นี้ว่า One-To-Many จากลูกค้าไปยังใบสั่งซื้อ
One-To-Many	รหัสใบสั่งซื้อ	รายการสั่งซื้อ	รหัสใบสั่งซื้อ	รายการสั่งซื้อ	เพราะว่าใบสั่งซื้อแต่ละใบสามารถสั่งสินค้าได้หลายชนิด เราจึงเรียกความสัมพันธ์แบบนี้ว่า One-To-Many
One-To-Many	สินค้า	รหัสสินค้า	รายการสั่งซื้อ	รหัสสินค้า	เพราะว่าสินค้า 1 ชนิดถูกสั่งซื้อได้หลายคน และถูกสั่งซื้อได้หลายครั้งด้วย เราจึงเรียกความสัมพันธ์นี้ว่า One-To-Many จากสินค้าไปยังรายการสั่งซื้อ

หลังจากที่เราได้กำหนดความสัมพันธ์แล้ว ขั้นตอนต่อไปเราจะมาพิจารณาว่าข้อมูลที่จะเก็บลงไปในแต่ละฟิลด์นั้นมีชนิดของข้อมูลเป็นอะไร ซึ่งชนิดของข้อมูลใน Microsoft Access นั้นมีทั้งหมด 9 ชนิดคือ

1. Text ใช้เก็บข้อมูลประเภทที่เป็นตัวอักษร เช่น ชื่อ นามสกุล
2. Memo ใช้เก็บข้อมูลที่เป็นคำอธิบายมาก ๆ และไม่มีรูปแบบ เช่น ฟิลด์ที่จะเก็บข้อมูลที่เป็นคำอธิบายหรือหมายเหตุ
3. Number ใช้เก็บข้อมูลที่เป็นตัวเลขทั่วไป
4. Date/Time ใช้เก็บข้อมูลที่เป็นเวลาหรือวันที่
5. Currency ใช้เก็บข้อมูลที่เป็นตัวเลขทางการเงิน
6. AutoNumber ใช้เก็บข้อมูลที่เป็นตัวเลขที่มีการเปลี่ยนแปลงอัตโนมัติและจะไม่ซ้ำกัน
7. Yes/No ใช้เก็บข้อมูล
8. OLE Object ใช้ในการเก็บรูปภาพและข้อมูลที่เป็น Object
9. LookUp Wizard ใช้ในการเชื่อมโยงกับตารางอื่น ๆ

หลักในการกำหนดชนิด ความยาว ทศนิยม ของ Field เราต้องกำหนดให้ถูกต้องกับข้อมูลจริง เช่น ถ้าเราต้องการกำหนดชนิดข้อมูลให้กับฟิลด์รหัสลูกค้าเราต้องดูว่า รหัสลูกค้า เราต้องการให้มีการเพิ่มขึ้นโดยอัตโนมัติหรือไม่ ถ้าหากต้องการเราก็กำหนดให้เป็น AutoNumber แต่ถ้าหากต้องการใช้รหัสลูกค้าอ้างอิงถึงข้อมูลต่าง ๆ ภายในรหัสเอง (เช่น รหัส 2 ตัวแรกหมายถึงปีที่เข้างาน 2 ตัวต่อมาหมายถึงฝ่ายหรือแผนก 3 ตัวสุดท้ายหมายถึงรหัสของพนักงาน) ถ้าเป็นข้อมูลในลักษณะนี้ควรเป็นตัวเลขผสมตัวอักษร แต่ถ้าเป็นฟิลด์ ชื่อ แน่นอนที่สุดชนิดข้อมูลต้องมีชนิดเป็น Text สำหรับความยาวนั้นต้องพิจารณาให้ดีว่าจะเก็บได้มากที่สุดกี่ตัวอักษร ถ้ามากเกินไปก็จะเปลืองเนื้อที่ในการจัดเก็บ แต่ถ้าหากน้อยเกินไปก็จะเก็บรายละเอียดได้ไม่หมด จากฟิลด์ต่าง ๆ ที่ได้ออกแบบมาผู้เขียนได้กำหนดชนิดและรายละเอียดต่าง ๆ ดังตารางที่ 3, 4, 5 และ 6

ตารางที่ 3

ลูกค้า				
ชื่อฟิลด์	ชนิด	ความยาว	ทศนิยม	หมายเหตุ
รหัสลูกค้า	AutoNumber	-	-	ไม่ต้องกำหนดความยาว เพราะ Access จะเพิ่มขึ้นเองอัตโนมัติ
ชื่อ	Text	25	-	
ที่อยู่	Text	30	-	
อำเภอ	Text	15	-	
จังหวัด	Text	15	-	
รหัสไปรษณีย์	Number	5	-	
โทรศัพท์	Text	30	-	เบอร์โทรศัพท์ต้องเป็น Text เพราะข้อมูลที่เก็บจะมีทั้งตัวเลขและตัวอักษร เช่น 123-4567 ต่อ 124 เป็นต้น

ตารางที่ 4

ใบสั่งซื้อ				
ชื่อฟิลด์	ชนิด	ความยาว	ทศนิยม	หมายเหตุ
รหัสใบสั่งซื้อ	AutoNumber	-	-	
รหัสลูกค้า	Number 6	0		
วันที่สั่งซื้อ	Date/Time	-	-	ข้อมูลชนิดวันที่ไม่ต้องกำหนดความยาว

ตารางที่ 5

สินค้า				
ชื่อฟิลด์	ชนิด	ความยาว	ทศนิยม	หมายเหตุ
รหัสสินค้า	AutoNumber	-	-	
สินค้า	Text	30	-	
ราคา	Number	8	2	เป็นเลข 8 หลักและมีทศนิยม 2 ตำแหน่ง

ตารางที่ 6

รายการสั่งซื้อ				
ชื่อฟิลด์	ชนิด	ความยาว	ทศนิยม	หมายเหตุ
รหัสใบสั่งซื้อ	Number	-	-	
รหัสสินค้า	Number	6	0	
จำนวน	Number	4	0	จำนวนที่สั่งซื้อ สามารถสั่งซื้อได้ไม่เกิน 9999 ชิ้น

จากเนื้อหาที่ได้กล่าวมา ผู้เขียนหวังเป็นอย่างยิ่งว่าคงจะช่วยให้ท่านผู้อ่านที่ยังไม่เข้าใจเรื่องฐานข้อมูลได้เข้าใจได้ดีขึ้น แต่อย่างไรเสียผู้อ่านควรจะศึกษาเพิ่มเติมจากหนังสือการออกแบบฐานข้อมูลทั่ว ๆ ไป ของบริษัทซีเอ็ดเองก็มีให้ศึกษา สำหรับในตอนต่อไปผู้เขียนจะพูดเกี่ยวกับการสร้างฐานข้อมูลที่ได้ออกแบบในตอนที่ 1 โดยใช้ Microsoft Access ถ้าหากผู้อ่านมีปัญหาไม่เข้าใจ หรือต้องการอยากสอบถามหรือพูดคุยอะไรเป็นการส่วนตัว คำแนะนำ หรือ ติชมอย่างไร ก็จดหมายหรือโทรศัพท์ติดต่อมาได้ที่

สิทธิศักดิ์ คล่องดี 355/76 ซ.ลาดพร้าว 122 ถ.ลาดพร้าว วังทองหลาง บางกะปิ กรุงเทพฯ 10310 โทร. 542-2266, 539-6992-3 ต่อ 116 (22.00-24.00 น.) หรือ บริษัท อินเตอร์เนชั่นแนล รีเสริท คอร์ปอเรชัน จำกัด 23/106-108 รอยเยลซิตี้ อเวนิว Block G ซ.ศูนย์วิจัย ถ.พระราม 9 ห้วยขวาง บางกะปิ กรุงเทพฯ 10310 โทร. 203-0009 ต่อ ฝ่ายวิจัยและพัฒนา (08.00-16.30 น. จันทร์-ศุกร์) *ผู้เขียนจักเป็นพระคุณอย่างยิ่ง*